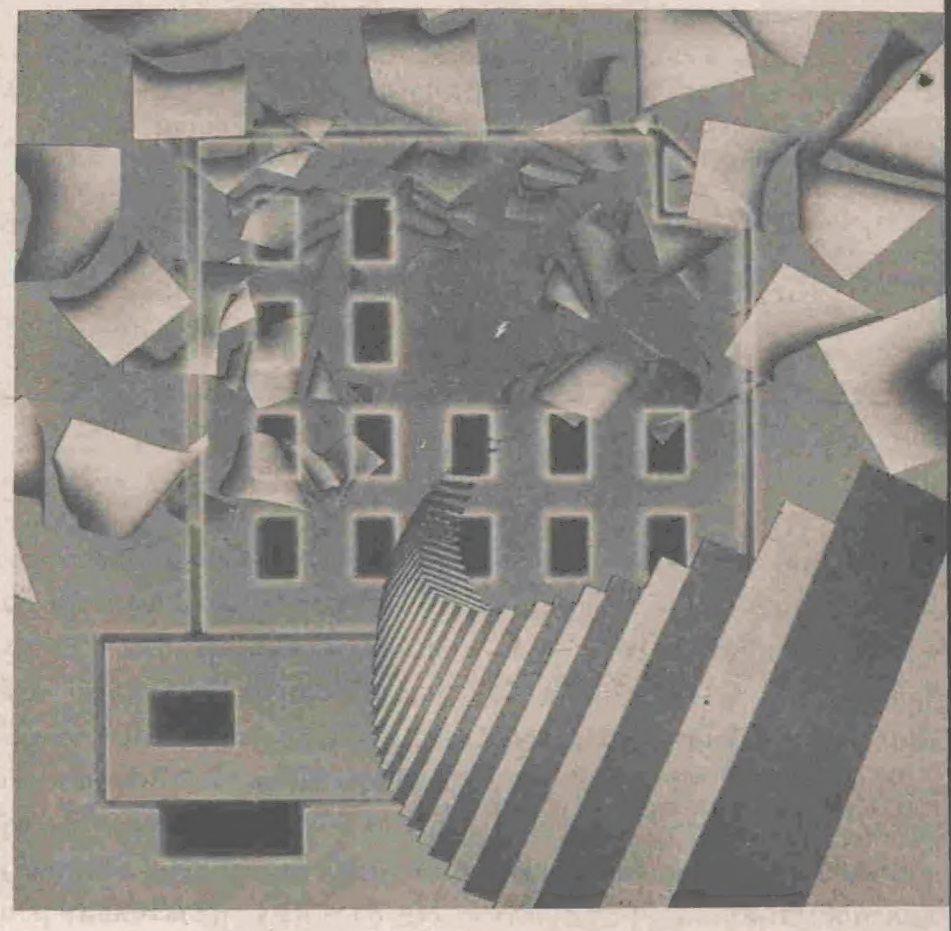
อาจหาญ สัตยารักษ์

SOFT MAIL

การพิมพ์อักขระ
แบบกราฟิก (ตอน 1)

วิธีการออกแบบอักขระกราฟิก
พิมพ์อักษรสูง 2 เท่า
พิมพ์สระบน สระล่าง
พิมพ์ตัวหนา ตัวขยาย
มีวิธีทำได้อย่างไร



จากบทความเรื่อง "โปรแกรมเพิ่มประสิทธิภาพเครื่องพิมพ์ 24 เข็ม" ในวารสารฯฉบับที่ 52 หน้า 264-271 นั้นเป็นการพิมพ์อักขระโดยอาศัย ROM ของเครื่องพิมพ์เป็นสำคัญ ผู้เขียนเห็นว่าการพิมพ์อักขระนั้นยังมีอีกวิธีหนึ่งที่สามารถจะทำได้โดยที่เครื่องพิมพ์ไม่จำเป็นต้องทำภาษาไทยไว้ก็ได้ นั่นคือ การพิมพ์แบบ bit image โดยต่อจากนี้ไปผู้เขียนจะขอใช้คำว่าการพิมพ์แบบกราฟิกแทน

เพื่อให้การศึกษาในเรื่องนี้เป็นไปอย่างสมบูรณ์ ผู้เขียนจึงจะขอแยกพูดเป็น 2 ตอน คือ

ตอนที่ 1 เป็นของเครื่องพิมพ์ 9 เข็ม

ตอนที่ 2 เป็นของเครื่องพิมพ์ 24 เข็ม

โดยเราจะเริ่มด้วยการพิมพ์ด้วยเครื่องพิมพ์ 9 เข็มกันก่อนในฉบับนี้

การพิมพ์แบบกราฟิก

การพิมพ์อักขระแบบกราฟิกนั้นเป็นการพิมพ์อักขระที่มีข้อดี คือ

1. ไม่ต้องพะวงว่าเป็นเครื่องพิมพ์ยี่ห้อใด
2. ไม่ต้องสนใจว่าได้ทำภาษาไทยในรอมของเครื่องพิมพ์แล้วหรือยัง
3. สามารถแก้ไขรูปแบบของตัวอักขระตามความต้องการของผู้ใช้ได้

เราจะมาดูตารางแสดงการพิมพ์แบบกราฟิกกันดังรูปที่ 1 ซึ่งจะเห็นว่าเราสามารถใช้การพิมพ์ได้หลายแบบ เช่น สมมติเราสร้างตัวอักขระ 1 ตัว ด้วยข้อมูล 22 ไบต์หากเราเลือกใช้โหมด 3 มาทำการพิมพ์ เราจะสามารถ

Option	จำนวนเข็ม	โหมด	จุด/นิ้ว	จุด/บรรทัด
Single–density	8	0	60	480
Double–density	8	1	120	960
High–speed Double–density	8	2	120	960
Quadruple–density	8	3	240	1,920
CRT I	8	4	60	640
Plotter graphics	8	5	72	576
CRT II	8	6	90	720
Dual density plotter graphics	8	7	144	1,152
Single–density	24	32	60	480
Double–density	24	33	120	960
CRT III	24	38	90	720
Triple–density	24	39	180	1,440
Hex–density	24	40	360	2,880

หมายเหตุ จุด/บรรทัดในตารางเป็นของกระดาษกว้าง 8 นิ้ว

รูปที่ 1 ตารางแสดงการพิมพ์แบบกราฟิก

พิมพ์อักขระใน 1 แถวได้ $\frac{1920}{22} = 87$ ตัว เป็นต้น

ดังนั้นการพิมพ์แบบกราฟิกนี้ เราจะเลือกใช้โหมดใดก็ได้ ซึ่งจะมีผลให้ได้ตัวอักขระที่มีรูปร่างต่างกันออกไป แต่ตัวพิมพ์ตามปกตินั้นเราจะใช้โหมด 3 เป็นหลักเสมอ เพราะเป็นตัวขนาดเล็กที่สุดเท่าที่จะพิมพ์ได้

การพิมพ์จากฟอนต์ที่สร้างไว้

เพื่อตัดความยุ่งยากในการสร้างรูปแบบตัวอักขระทั้งภาษาไทยและภาษาอังกฤษ ผู้เขียนจึงขอใช้สิ่งที่มีอยู่แล้ว คือ รูปแบบของตัวอักขระที่มีอยู่ในราชวิถีเวิร์ดพีซีมาใช้เลย จึงต้องขอขอบคุณนายแพทย์ชุษณะ มกรสาร มา ณ ที่นี้อีกครั้ง

ขอให้ผู้อ่านย้อนกลับไปอ่านบทความเรื่อง "การสร้างอักขระกราฟิกด้วยเวิร์ดพีซี" ในวารสารไมโครฯ ฉบับที่ 52 หน้า 279-285 อีกที เพื่อจะได้เข้าใจถึงการเก็บอักขระในไฟล์ว่าเป็นอย่างไร

การพิมพ์อักขระด้วยเครื่องพิมพ์ 9 เข็มเรามีหลักดังนี้

1. จัดสร้างอักขระที่ต้องการพิมพ์ก่อน และเพื่อให้การพิมพ์เป็นไปอย่างสวยงามเราก็สร้างไว้เป็น 2 ชุดต่อหนึ่งอักขระ (โดยจะใช้เทคนิคของการเลื่อนกระดาษขึ้นเล็กน้อยหลังการพิมพ์ข้อมูลชุดที่ 1 แล้วพิมพ์ข้อมูลชุดที่ 2 ตาม) จึงทำให้ข้อมูลมี 44 ไบต์ต่อหนึ่งอักขระ

2. ข้อมูลทั้งสองชุดนี้ถูกจัดเก็บลงในแฟ้มข้อมูลต่อเนื่องกันไปตั้งแต่ข้อมูลลำดับที่ 0 ถึง 223 ทั้งนี้ให้ข้อมูลลำดับที่ 0 เป็นรหัสแอสกี 32 และข้อมูลลำดับที่ 223 คือ

3. เมื่อต้องการพิมพ์ข้อความก็คำนวณหาจำนวนครั้งการสั่งพิมพ์ที่จะใช้ในหนึ่งบรรทัด (หมายถึงในหนึ่งบรรทัดเป็นแบบการพิมพ์ภาษาอังกฤษ) ซึ่งมีค่าเท่ากับจำนวนอักขระที่อยู่ระดับกลางทั้งหมดนำมาคูณด้วย 22 ก็จะได้จำนวนครั้งทั้งหมดที่จะไปกำหนดให้เครื่องพิมพ์รับรู้ก่อนการพิมพ์แบบกราฟิก

4. ขณะที่พิมพ์ก็ส่งข้อมูลที่ตรวจสอบได้จากรหัสแอสกีนั้นไปพิมพ์แบบกราฟิกตามข้อมูลที่อ่านได้จากตัวแปร Font ลำดับที่คำนวณได้ ซึ่งจะทำให้การพิมพ์ได้ผลตรงตามรหัสที่เก็บไว้

เพื่อสะดวกต่อการศึกษาผู้เขียนได้จัดเป็นซอฟต์เมล์ด้วย โดยใช้ชื่อไฟล์ว่า BITPRINT.PAS และมีเพียงหนึ่งไฟล์เท่านั้น โดยมีรายละเอียดดังนี้

โปรแกรมเริ่มต้นของเราจะเป็นดังรูปที่ 2 ซึ่งมีชุดคำสั่ง 2 ชุด คือ ReadFont เพื่อทำการอ่านรูปแบบของอักขระมาเก็บไว้ในตัวแปร Font1[0...223] และ Font2[0...223] กับชุดคำสั่ง CopyByte เพื่อที่จะส่งรหัสไปให้แก่เครื่องพิมพ์

```
PROGRAM PrintBitImage;
uses crt,dos;
Type Font=array [1..44] of byte;
var
     P_Big,P_mode,Boldck,Enlarge,double,Underline,
     RoundP,Leftmargin,FontMode,Inver,FeedIn,FeedOut:byte;
     Font1,Font2:array[0..223] of Font;
     FontDisp:Font;
     Regs : Registers;

PROCEDURE ReadFont;
var   size:word;  FontFile:File;
 BEGIN
  clrscr; textcolor(0); TextBackground(7);
  writeln('                                          ');
  writeln('            Print by Bit Image            ');
  writeln('  By Arthan Sattayarak  Lampang 52000     ');
  writeln('                                          ');
  textcolor(3); TextBackground(0);  Gotoxy(1,7);
  writeln('Loading FONT1.NLQ');
     Assign(FontFile,'B:font1.nlq');
     Reset(FontFile,1);
     size:=Filesize(FontFile);
     blockRead(FontFile,Font1,size);
     Close(FontFile);
  writeln('Loading FONT2.NLQ');
     Assign(FontFile,'B:font2.nlq');
     Reset(FontFile,1);
     size:=Filesize(FontFile);
     blockRead(FontFile,Font2,size);
     Close(FontFile);
end; { ReadFont }

PROCEDURE CopyByte(Bytest : string);
 var  B: byte;
  begin
     for b:=1 to length(ByteSt) do
      begin
       Regs.AH:=0;          Regs.AL := ord(ByteSt[b]);
       Regs.DX := Pred(1);  Intr($17, Regs);
       end;
  end;
```

รูปที่ 2 ชุดคำสั่งอ่านข้อมูลอักขระกราฟิ
และชุดคำสั่งควบคุมเครื่องพิมพ์

ในชุดคำสั่ง CopyByte นั้น ถ้าเปลี่ยน Pred(1); ไปเป็น Pred(2); ก็จะเป็นการสั่งให้เครื่องพิมพ์เครื่องที่ 2 ที่ต่อกับเครื่องคอมพิวเตอร์ทำการพิมพ์ข้อความที่ส่งไปให้

การพิมพ์อักขระตามปกติ

ในการพิมพ์ตัวอักขระตามปกตินั้นจะเป็นไปดังชุดคำสั่ง CopyOneChar (ในรูปที่ 3) โดยจะส่งข้อมูลไบต์ที่ 1 - 22 ไปพิมพ์ก่อนเมื่อค่า PRound ที่ส่งมามีค่าเป็น 1 และพิมพ์ข้อมูลไบต์ที่ 23 - 44, พิมพ์อีกครั้งเมื่อค่า PRound ที่ส่งมามีค่าเป็น 2 (ค่าตัวแปร BitP เปลี่ยนจาก 0 ไปเป็น 22)

เมื่อกำหนดให้พิมพ์ตัวหนา (ค่าตัวแปร Boldck = 1) ก็ให้นำไบต์ที่อยู่ก่อนหน้านี้ 1 ไบต์มารวมกับแบบ OR ซึ่งจะมีผลทำให้ได้ตัวหนาปรากฏบนกระดาษได้

ในกรณีที่มีการขีดเส้นใต้ (ค่าตัวแปร UnderLine = 1) และเป็นการพิมพ์กลุ่มข้อความที่เป็นสระต่าง ๆ คือ ุ ู ก็ให้ทำการรวมข้อมูลนั้น ๆ กับ 64 โดยการรวมแบบ OR เช่นกัน ซึ่งจะมีผลทำให้หัวเข็มที่คุมด้วยบิตที่ 6 ทำงาน จึงได้เส้นใต้ตามต้องการ การขีดเส้นใต้นี้ผู้เขียนยังไม่ได้ตรวจสอบการเว้นเมื่อเป็น ุ ู ญ ฎ ฏ ฐ ฤ ก็ให้ผู้อ่านลองแก้ไขดูเองด้วย

หากมีการกำหนดให้พิมพ์ตัวขยาย (ค่าตัวแปร Mode = 1) นั้นจะใช้วิธีส่งรหัสไปพิมพ์อีกหนึ่งครั้งก็จะได้ตัว

```
PROCEDURE CopyOneChar(PRound,Mode:byte);
 var   OneByte,B,BitP:byte;
  begin
    BitP:=0; if PRound=2  then BitP:=22;
    for b:=1 to 22 do
     begin
       OneByte:=FontDisp[b+BitP];
       if (Boldck=1) and (b>1) then OneByte:=OneByte or FontDisp[b+BitP-1];
       if (UnderLine=1) and (RoundP=2) then OneByte:=OneByte or 64;
       if Inver=1 then OneByte:=OneByte xor 255;
       copyByte(chr(OneByte));
       if Mode=1 then copyByte(chr(OneByte));
     end;
 end;

PROCEDURE CopyOneHighChar(PRound,Mode,RR:byte);
const BitHigh:array[0..7] of byte=(3,12,48,192,3,12,48,192);
      Bitck:array[0..7] of byte=(1,2,4,8,16,32,64,128);
 var   OneByte,B,BitP,x,Highbyte,Stck,EndCk:byte;
  begin
    BitP:=0; if RR=2 then BitP:=22;
    Stck:=4; Endck:=7; If Pround=2 then begin Stck:=0; Endck:=3; end;
    for b:=1 to 22 do
      begin
       OneByte:=FontDisp[B+BitP]; Highbyte:=0;
       if (Boldck=1) and (b>1) then OneByte:=OneByte or FontDisp[b+BitP-1];
       if (UnderLine=1) and (RoundP=2) and (RR=2) then
                                          OneByte:=OneByte or 64;
       for x:=Stck to Endck do
            if OneByte and Bitck[x]=Bitck[x] then
                                HighByte:=HighByte+BitHigh[x];
         copyByte(chr(HighByte));
         if Mode=1 then copyByte(chr(HighByte));
      end;
  end;
```

รูปที่ 3 ชุดคำสั่งพิมพ์อักขระปกติ และพิมพ์แบบสูง 2 เท่า

ขยายตามต้องการ (จะต้องมีการพิมพ์) ข้อมูลเป็น 2 เท่าทุกครั้งที่มีตัวขยาย โดยจะอธิบายในภายหลัง)

การพิมพ์ตัวกลับขาวดำก็ทำได้โดยการให้รวมค่าที่ได้กับค่า 225 แบบ XOR

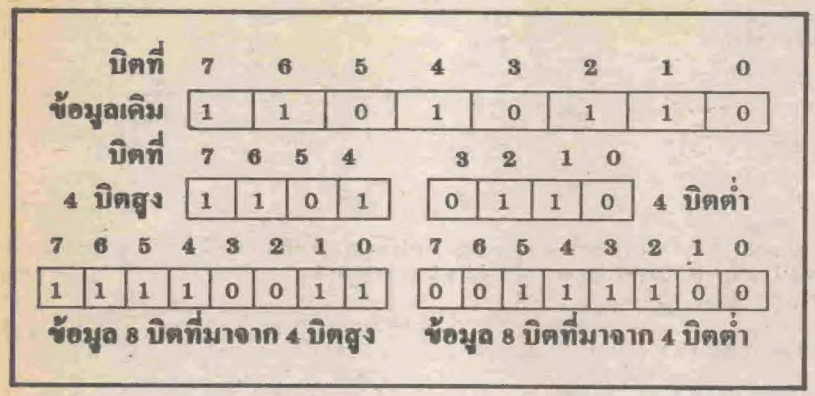
การพิมพ์อักขระสูง 2 เท่า

เราสามารถใช้ข้อมูลตามปกตินี้มาพิมพ์ให้เป็นตัวอักษรสูง 2 เท่าได้ โดยไม่ต้องแก้ไขข้อมูลแต่อย่างใดทั้งสิ้น เพราะเราทราบแล้วว่าการพิมพ์หนึ่งครั้งเครื่องพิมพ์ต้องการข้อมูล 8 บิตเพื่อกำหนดให้เข็ม 8 เข็มทำงาน ดังนั้นเมื่อเรามีข้อมูล 8 บิตขนาดสูงปกติ อยากจะทำให้เป็นตัวสูง 2 เท่าก็จะทำได้ดังนี้

1. นำข้อมูล 8 บิตมาแยกเป็น 2 ส่วน คือ 4 บิตสูง และ 4 บิตต่ำ

2. นำข้อมูล 4 บิตสูงมาขยายให้เป็นข้อมูล 8 บิต ส่งไปพิมพ์ครั้งที่ 1 จะได้ตัวอักขระครึ่งบน ซึ่งจะสูงเท่ากับตัวอักขระที่พิมพ์ตามปกติ แล้วเลื่อนกระดาษขึ้น

3. นำข้อมูล 4 บิตต่ำมาขยายให้เป็นข้อมูล 8 บิต ส่งไปพิมพ์เป็นครั้งที่สอง ก็จะได้ตัวอักขระครึ่งล่าง ก็จะได้ตัวอักขระสูง 2 เท่าตามต้องการ



รูปที่ 4 การขยายข้อมูล 8 บิตให้กลายเป็นข้อมูล 16 บิต

ดังตัวอย่างในรูปที่ 4 (ในกรณีของเครื่องพิมพ์ให้มองข้อมูลนี้ในแนวตั้ง โดยมีบิตที่ 7 อยู่บนสุดและบิตที่ 0 อยู่ล่างสุด)

การทำงานจริง ๆ ก็คือ ชุดคำสั่ง CopyOneHighChar (ในรูปที่ 3) โดยจะทำการตรวจสอบข้อมูล 8 บิต ครั้งละ 4 บิต โดยใช้ตัวแปร Stck และ Endck ซึ่งจะมีค่า 4–7 ครั้งหนึ่ง (4 บิตสูง) และมีค่า 0–3 อีกครั้งหนึ่ง (4 บิตต่ำ) สลับกันไป ซึ่งจะได้ตัวแปร HighByte เป็นข้อมูล 8 บิตมาใช้สั่งเครื่องพิมพ์ทำงาน

สังเกตจากโปรแกรมจะเห็นว่าถ้าบิตที่ 0 ของตัวแปร OneByte เป็น 1 ก็จะเพิ่มค่าให้แก่ตัวแปร HighByte อีก 3 หรือถ้าบิตที่ 3 เป็น 1 ก็จะเพิ่มอีก 192 เป็นต้น คือเป็นการเทียบตารางระหว่างตัวแปรมิติ Bitck [0..7] และ BitHigh [0..7] นั่นเอง

ในชุดคำสั่งของการพิมพ์ขนาดสูง 2 เท่านี้ยังสามารถพิมพ์แบบตัวหนาและขีดเส้นใต้ได้ แต่จะพิมพ์แบบตัวกลับขาวดำไม่ได้ ถ้าต้องการก็ให้ลองแก้ไขเอาเองนะครับ

การพิมพ์ในหนึ่งบรรทัด

การพิมพ์แบบกราฟิกนี้เราจะต้องรู้จำนวนครั้งที่จะสั่งให้เครื่องพิมพ์ทำงาน ดังนี้การตรวจสอบในการพิมพ์นี้จึงจำเป็นมาก เช่น ตัวปกติเราสั่งพิมพ์ 22 ครั้งต่อหนึ่งอักขระ แต่ถ้าตัวขยายเราจะต้องสั่งพิมพ์เป็น 44 ครั้งต่อหนึ่งอักขระ เป็นต้น ดังนั้นจึงต้องตรวจข้อความไปตั้งแต่ต้นจนสุดบรรทัดว่ามีกี่ตัวอักขระ โดยคิดว่าถ้ามีตัวขยายก็คือมีตัวอักขระเพิ่ม 1 ตัวต่ออักขระหนึ่งตัวเสมอ ขอให้ดูชุดคำสั่ง CopyOneLine (รูปที่ 5) จะใช้ตัวแปร cc เป็นตัวตรวจสอบว่าเป็นตัวขยายหรือไม่ ถ้าเป็นก็เพิ่มค่าตัวแปร Count ทีละ 2 (ตามปกติจะเพิ่มทีละ 1 เท่านั้น)

ตัวแปร P_mode นั้นใช้ควบคุมขนาดตัวอักษรซึ่งในกรณีนี้จะใช้รหัส #3(^PC), #12(^PL), #17(^PQ) และ #23(^PW) เป็นตัวกำหนด

การตรวจข้อความถ้าพบรหัสแอสกีน้อยกว่า 32 ก็จะใช้เป็นตัวควบคุมลักษณะของตัวอักขระ ได้แก่

#2(^PB) ใช้เข้า–ออกตัวหนา

#5(^PE) ใช้เข้า–ออกตัวขยาย

#18(^PR) ใช้เข้า–ออกตัว
กลับขาวดำ

#19(^PS) ใช้เข้า–ออกการ
ขีดเส้นใต้

#28(^P\, ^P1) เลือกใช้ fon
ที่ 1

#29(^P], ^P2) เลือกใช้ fon
ที่ 2

การพิมพ์ในแต่ละบรรทัดนั้น เมื่อ
พิมพ์รอบแรกจบลงจะทำการเลื่อนกระ
ดาษขึ้นไป $\frac{1}{216}$ นิ้วก่อนจึงจะพิมพ์รอบ
ที่สอง คือคำสั่ง CopyByte (#27# #51
#1#10#27#51+chr(FeedIn)); ทั้งนี้
เพื่อให้ข้อมูลที่เก็บไว้ได้พิมพ์เหลื่อมกัน
ซึ่งจะทำให้ได้ผลการพิมพ์ที่สวยงาม
รายละเอียดอื่น ๆ ของชุดคำสั่งนี้กรุณา
ศึกษาด้วยตนเองให้เข้าใจด้วย

การจัดภาษาไทย ให้เครื่องพิมพ์

โดยแนวคิดเดียวกันกับการพิมพ์
โดยอาศัยตัวอักขระในรอม (ฉบับที่ 52
ดังกล่าวข้างต้น) ก็ใช้วิธีการเดียวกัน
กับชุดคำสั่ง CopyFont (รูปที่ 2 หน้า
267 ฉบับที่ 52) ในที่นี้ผู้เขียนเปลี่ยน

```
PROCEDURE CopyOneLine(Content:string);
 var k,R,n1,n2,Count:word;   cc,R_round:byte;
begin
  count:=0; cc:=0;
  For K:=1 to length(Content) do
     begin
        case content[k] of
                #5:Inc(cc);
          #32..#255:if cc mod 2=0 then Inc(Count) Else Inc(Count,2);
        end;
     end;
      N1:=lo(succ(Count*22-1)); n2:=Hi(succ(Count*22-1));
 for R:=1 to 2 do
  begin  R_Round:=1;
   Repeat
     CopyByte(#27#51+chr(FeedIn));
     case P_mode of
        0:CopyByte(#27'*'#3+chr(n1)+chr(n2));
        3:CopyByte(#27'*'#7+chr(n1)+chr(n2));
       12:CopyByte(#27'*'#0+chr(n1)+chr(n2));
       17:CopyByte(#27'*'#4+chr(n1)+chr(n2));
       23:CopyByte(#27'*'#1+chr(n1)+chr(n2));
     end;
    for k:=1 to Length(Content) do
    begin
     case Content[k] of
      #2:Begin Inc(Boldck); Boldck:=Boldck mod 2; end;
      #5:begin Inc(Enlarge); Enlarge:=Enlarge mod 2; end;
     #18:begin Inc(Inver); Inver:=Inver mod 2; end;
     #19:begin Inc(UnderLine); UnderLine:=UnderLine mod 2; end;
     #28:FontMode:=1;
     #29:FontMode:=2;
    #32..#255:begin
               if FontMode=1 then Fontdisp:=Font1[ord(content[k])-32]
                  Else Fontdisp:=Font2[ord(content[k])-32];
               if P_big=1 then CopyOneHighChar(R,Enlarge,R_round)
                          Else CopyOneChar(R,Enlarge);
              end;
     end;  { end case }
    end;  { end k }
       Copybyte(#27#51#1#10#27#51+chr(FeedIn));
       inc(R_Round); If P_big=0 then R_Round:=3;
   Until  R_Round>2;
     if P_big=1 then  begin
       if r=1 then  copyByte(#10)
              Else if RoundP<>2 then copyByte(#27#51#14#10)
                   Else copyByte(#27#51+chr(FeedOut)+#10);
                    end;
 end;  {end r}
     if P_big=0 then begin
         if RoundP=2 then CopyByte(#27#51+chr(FeedOut)+#10)
                  Else CopyByte(#10);
                    end;
end;
```

รูปที่ 5 ชุดคำสั่งสำหรับจัดการพิมพ์ใน 1 บรรทัด (ระดับเดียว)

แนวการเขียนไปอีกแบบหนึ่ง แต่แนวการอธิบายยังคงคล้ายของเดิม และขอใช้ชื่อชุดคำสั่ง CopyFont เช่นเดียวกัน รายละเอียดดูในรูปที่ 6 ซึ่งคิดว่าคงทำความเข้าใจได้ไม่ยากเช่นกัน

ส่วนของตาราง #171– #174 คือ อักขระ ญ ฎ #149– #159 คือ อักขระแสดง ฎ ฐ

```
PROCEDURE CopyFont(Content:string);
Var PWORD:array[0..2] of string;       R:word;

    Procedure SetCode(Code:char);
      var x:byte;
      begin
        for x:=0 to 2 do Pword[x]:=PWord[x]+Code;
      end;
    Procedure Set_Code(Code:string);
      var x:byte;
      begin
        for x:=0 to 2 do
            if Code[x+1]<>#0 then Pword[x]:=PWord[x]+Code[x+1];
      end;
begin
    For R:=0 to 2 do Pword[R]:='';
    P_Big:=0;  P_mode:=0;  R:=1;
 Repeat
   Case Content[R] of
     #2,#5,#18,#19,#28,#29:setCode(Content[R]);
     #3,#12,#17,#23:P_mode:=ord(Content[R]);
     #8:P_Big:=1;
     #32..#127,#158,#160..#170,#175..#193,#195..#206,#208,
                      #209,#213,#214:set_code(#32+Content[R]+#32);
     #149..#151:set_code(#32+content[R]+#159);
     #152..#154:set_code(#159+Content[R]+#159);
     #155..#157:set_code(#159+Content[R]+#32);
     #159:set_Code(#159+#159+#159);
     #171..#174:set_code(#32+Content[R]+chr(ord(Content[R])-36));
     #194:set_code(#32+Content[R]+#139);
     #207:begin Delete(Pword[0],length(Pword[0]),1);
                PWord[0]:=Pword[0]+#222+' ';
                set_code(#0+#206+#32);
          end;
     #210..#212:set_code(chr(ord(Content[R])-78)+#131+#32);
     #215,#216:begin Delete(Pword[2],length(Pword[2]),1);
                     set_Code(#0+#0+Content[R]);
               end;
     #217..#255:begin Delete(Pword[0],length(Pword[0]),1);
                      Set_code(Content[R]+#0+#0);
                end;
   end; { end case }
     Inc(R);
 Until R>Length(Content);
    CopyByte(#27#108+chr(leftMargin));
    for RoundP:=0 to 2 do CopyOneLine(PWord[RoundP]);
END;
```

รูปที่ 6 ชุดคำสั่งจัดภาษาไทยให้เป็น 3 บรรทัดเพื่อส่งไปพิมพ์

#194 คือ อักขระ ฤ

#207 คือ สระ ำ

#215– #216 คือ สระ เ เ

#217– #255 คือ สระบน และวรรณยุกต์

การจัดสระและวรรณยุกต์ของภาษาไทยในหน่วยความจำให้ตรงกับการเก็บรูปแบบอักขระในการพิมพ์ก็ยังคงคล้ายกับของเดิม (ชุดคำสั่ง Print—Font รูปที่ 4 หน้า 269 ฉบับที่ 52) แต่คราวนี้ขอใช้ชื่อชุดคำสั่ง Print แทน ซึ่งดูรายละเอียดได้จากรูปที่ 7 และขอให้อ่านคำอธิบายจากฉบับที่ 52 เพิ่มเติม

ถึงตรงนี้เราก็ได้ชุดคำสั่งที่ใช้งานสมบูรณ์พร้อมที่จะทำงานได้แล้ว โดยเก็บชุดคำสั่งเรียงลำดับตามที่ได้กล่าวมาแต่ต้น จากนี้เราก็จะลองเขียนข้อความส่งไปให้พิมพ์ดูกันสัก 2 ชุด

รูปที่ 7 ชุดคำสั่งจัดระดับภาษาไทยให้ถูกต้องกับการพิมพ์ด้วยเครื่องพิมพ์

```
PROCEDURE Print(Content:string);
var K:byte; WordSt:string;
begin   WordSt:='';  content:=content+' ';
 for k:=1 to Length(Content)-1 do
  begin
   if content[k] in [#1..#206,#208..#216,#222,#223,#228] then
                                WordSt:=WordSt+Content[k];
   if content[k]=#207 then begin
          if content[k-1] in [#224..#227] then
              WordSt:=WordSt+#206 else  WordSt:=WordSt+Content[k];
                              end;
  if content[k] in [#224..#227] then      begin
      if content[k+1]=#207 then WordSt:=WordSt+chr(ord(content[k])+6)
       ELSE WordSt:=WordSt+Content[k];    end;
  if (content[k] in [#217..#221])  then          begin
  if content[k+1] in [#1..#223] then WordSt:=WordSt+Content[k];
  if content[k+1] in [#224..#228] then   begin
    case content[k] of
     #217 :Begin WordSt:=WordSt+chr(ord(content[k+1])+14); Inc(k); end;
     #218 :Begin WordSt:=WordSt+chr(ord(content[k+1])+19); Inc(k); end;
     #219 :Begin WordSt:=WordSt+chr(ord(content[k+1])+23); Inc(k); end;
     #220 :Begin WordSt:=WordSt+chr(ord(content[k+1])+27); Inc(k); end;
     #221 :Begin WordSt:=WordSt+chr(ord(content[k+1])+10); Inc(k); end;
    end;                                  end;      end;
 end;
      CopyFont(WordSt);
end;
```

การส่งข้อความไปพิมพ์

เราสามารถให้พิมพ์ข้อความได้ทั้งการกำหนดในปาสคาลเอง หรืออ่านมาจากที่พิมพ์ด้วยเวิร์ดตัวอื่น (ในที่นี้ผู้เขียนจะไม่กล่าวถึง ถ้าต้องการใช้ก็ขอให้ย้อนไปอ่านฉบับที่ 52)

ขอให้ดูรูปที่ 8 ซึ่งมี 3 ชุดคำสั่ง คือ ชุดคำสั่ง setPrinter, Print1 และ Print2 อันเป็นชุดคำสั่งที่กำหนดตัวแปรเริ่มต้นในการพิมพ์, ข้อความทดลองพิมพ์ 2 ชุด รหัสตามที่เห็นเช่น

รูปที่ 8 ชุดคำสั่งทดลองพิมพ์ข้อความ 2 ชุด

```
PROCEDURE Print1;
  begin
    Print(#8#3#2#19'การพิมพ์อักขระแบบ '#5'Bit'#5' '#5'Image'#19#5#2);
    Print('        '#3#2'สำหรับเครื่องพิมพ์ 9 เข็ม'#2);
    Print(#8#29'          พัฒนาโดย '#28#5'อาจหาญ สัตยารักษ์'#5);
    Print(#8'            222 พหลโยธิน อ.เมือง  ลำปาง 52000');
    Print(#23#5'รหัส'#5'ทิ.ใช้ควบคุม '#29'คือ'#28);
    Print(#2'     ^PB ตัวหนา'#2'                '#5'^PE ตัวขยาย'#5);
    Print('     '#19'^PS ขีดเส้นใต้'#19'                '#18#5+
                                        '^PR ตัวกลับขาวดำ'#5#18);
    Print('     ^P\(^P1) Font ทิ. 1      '#29'^P](^P2) Font ทิ. 2'#28);
    Print(#8'     ^PHตัวสูง 2 เท่า');
  end;

PROCEDURE Print2;
  begin
    Print('ตัวพิมพ์ปกติขนาดธรรมดา และ'#5'ตัวขยาย '#29'ตัวเอียงขยาย'#28#5
    Print(#3'^PC ตัวกว้างแบบทิ. 1 '#5'ตัวขยาย'#29'ตัวเอียง'#28#5);
    Print(#23'^PW ตัวกว้างแบบทิ.2 '#5'ตัวขยาย'#29'ตัวเอียง'#28#5);
    Print(#17'^PQ กว้างทิ.3 '#5'ขยาย'#29'เอียง'#28#5);
    Print(#12'^PLกว้าง4'#29'เอียง'#28#5'ขยาย'#5);
    Print(#8'ตัวสูง 2 เท่า(ขนาดปกติ) '#5'ตัวขยาย '#29'ตัวเอียง'#28#5);
    Print(#8#3'ตัวสูงกว้างแบบทิ.1 '#5'ตัวขยาย'#29'ตัวเอียง'#28#5);
    Print(#8#23'ตัวสูงกว้าง2 '#5'ตัวขยาย'#29'ตัวเอียง'#28#5);
    Print(#8#17'สูงกว้าง3'#5'ขยาย'#29'ตัวเอียง'#28#5);
```

รูปที่ 8 (ต่อ)

```
  Print(#8#12'สูง'#5'กว้าง'#5'4'#29#5'เอียง'#28#5);
  end;

Procedure setPrinter;
  begin
   Boldck:=0;        Enlarge:=0;      double:=0;
   Underline:=0;     FontMode:=1;     Inver:=0;
   LeftMargin:=4;    FeedIn:=21;      FeedOut:=12;
  end;

{ ==================== MAIN ====================}
 begin
    ReadFont;        setPrinter;
    Writeln;
    Writeln('                                    ');
    writeln('              กำลังพิมพ์ลงกระดาษ ');
    Print1;   Print2;
 end.
```

รูปที่ 9 ผลที่ได้จากการพิมพ์ของชุดคำสั่ง Print1

การพิมพ์อักขระแบบ Bit Image

สำหรับเครื่องพิมพ์ 9 เข็ม

พัฒนาโดย อาจหาญ สัตยารักษ์

222 พหลโยธิน อ.เมือง ลำปาง 52000

รหัสที่ใช้ควบคุม คือ

^PB ตัวหนา ^PE ตัวขยาย

^PS ขีดเส้นใต้ ^PR ตัวกลับขาวดำ

^P\(^P1) Font ที่ 1 ^P]\(^P2) Font ที่ 2

^PHตัวสูง 2 เท่า

ตัวพิมพ์ปกติขนาดธรรมดา และตัวขยาย ตัวเอียงขยาย

^PC ตัวกว้างแบบที่ 1 ตัวขยายตัวเอียง

^PW ตัวกว้างแบบที่2 ตัวขยายตัวเอียง

^PQ กว้างที่3 ขยายเอียง

^PL กว้าง4 เอียงขยาย

ตัวสูง 2 เท่า(ขนาดปกติ) ตัวขยาย ตัวเอียง

ตัวสูงกว้างแบบที่1 ตัวขยายตัวเอียง

ตัวสูงกว้าง2 ตัวขยายตัวเอียง

สูงกว้าง3 ขยายตัวเอียง

สูงกว้าง4 เอียง

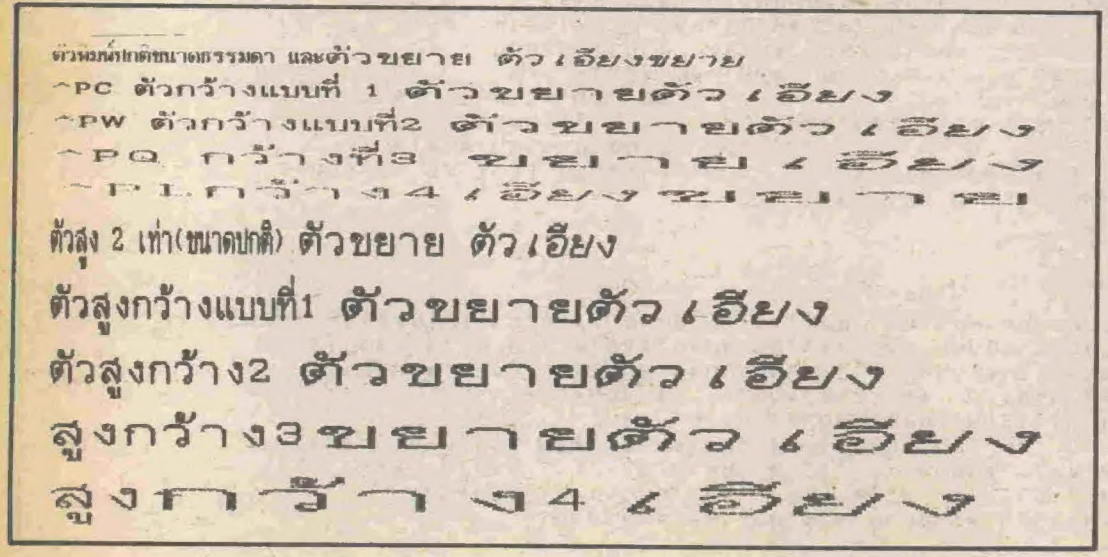
#3 สามารถป้อนในปาสคาลเป็น ^PC เลยก็ได้

รูปที่ 9 เป็นผลจากการพิมพ์ของข้อความในชุดคำสั่ง Print1 และรูปที่ 10 เป็นผลจากการพิมพ์ในชุดคำสั่ง Print2

ถึงตรงนี้ผู้อ่านก็คงได้รับรู้การพิมพ์แบบกราฟิกซึ่งให้ผลการพิมพ์ได้หลายแบบ ทั้งนี้ขึ้นกับแนวคิดที่เราดำเนินการด้วย หวังว่าคราวนี้คงจะได้พิมพ์ข้อความที่มีหน้าตาแปลก ๆ กันสนุกไปเลย

ตัวแปร FeedIn เป็นการตั้งระยะห่างของสระบน, สระล่าง ส่วน FeedOut เป็นการตั้งระยะห่างระหว่างบรรทัดของภาษาไทย ให้ลองเปลี่ยนค่าดู

พบกันใหม่คราวหน้า ในตอนที่ 2 สำหรับผู้มีเครื่องพิมพ์แบบ 24 เข็มครับ...



รูปที่ 10
ผลที่ได้จากการพิมพ์ของชุดคำสั่ง Print2